Alexander Garcia June 5, 2017

Assignment-2

1. (a) Consider the functions $f(x) = x^3 - 2,\ g(x) = e^x - 5sin(x^3) - 3cos(x)$.

   The initial step in the bisection process is to choose a range of x in which $f(x) = 0$. As the range is given to be $(0, 2)$, the bisection algorithm proceeds by guessing the midpoint of the range (let's call this $x_0 = \frac{b_0 - a_0}{2}$) to be a zero of the function. If $|f(x_0)| \leq 0 + tol$, then the zero is said to have been found. If $|f(x_0)| > 0 + tol$, then either a_0 or b_0 is replaced by x_0, depending on where $f(x)$ changes sign. The process is then repeated until $f(x_n) < 0 + tol$.

   At each stage in the algorithm, the maximum possible error of $f(x_n)$ is $e_n = \frac{b_n - a_n}{2}$, and the general error is $e_n \leq \frac{b_n - a_n}{2}$. Because $b_n - a_n = \frac{b_{n-1} - a_{n-1}}{2}$, we can express the current error as

   $$e_n \leq \frac{1}{2} e_{n-1}$$

   In this case we are taking 34 iterations until e_n is reasonably low, so $e_{34} = \frac{1}{2} e_{33}$. It is clear that this expression will cascade down to e_0, the error of the initial guess. The final error e_{34} can therefore be generalized to

   $$e_0 * \prod_{i=1}^{34} \frac{1}{2} = 1 * \prod_{i=1}^{34} \frac{1}{2} = \frac{1}{2^{34}}$$

   This result shows that the errors of the bisection method do not rely at all upon the function whose zeroes are being calculated, but rather solely upon the number of iterations carried out by the process. The number of iterations to calculate $g(x) = 0$ to sufficent accuracy is exactly the same number of iterations needed to calculate $f(x) = 0$.

   (b) The convergence rate of a root finding algorithm is the limit of the ratio of the error in iteration n to the error in iteration $n - 1$ as $n \to \infty$.

   $$conv = \lim_{n \to \infty} \frac{e_n}{e_{n-1}}$$

   (i) This algorithm is linearlly convergent, or has a constant convergance rate, as the ratio between every adjacent error is $\frac{1}{2}$.

   (ii) This algorithm is quadratically convergent. It is clear that the ratio of e_n to e_{n-1} is cut in half after every iteration, making the rate of convergance not a constant factor, but a result of which iteration is currently taking place.

   $$(\frac{e_n}{e_{n-1}} = \frac{1}{2^n})|_{n=1}^{\infty}$$

(c) For a “good” starting guess when undergoing Newton approximation, the rate of convergence is quadratic.

After choosing a good starting guess, the next approximations for $f(x) = 0$ are determined by $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$, where x_n approaches the true zero, $x = \alpha$. At each stage, the error is $e_n = |x_n - \alpha|$, and the rate of convergance would be $\frac{e_n}{e_{n+1}}$.

We let the function $f(x)$ be approximated by the first order Taylor polynomial about x_n.

$$f(x) = f(x_n) + (x - x_n)f^{'}(x_n) + \frac{1}{2}(x - x_n)^2 f^{''}(c_n)$$

where c_n is a constant close to x_n used to estimate the remainder. If we evaluate the approximation at $x = \alpha$, we would ideally get zero, as this is the true root of the function.

$$0 = f(x_n) + (\alpha - x_n)f^{'}(x_n) + \frac{1}{2}(\alpha - x_n)^2 f^{''}(c_n)$$

If we then divide by $f^{'}(x_n)$, we see that the expression for x_{n+1} appears.

$$0 = \frac{f(x_n)}{f^{'}(x_n)} + (\alpha - x_n) + \frac{(\alpha - x_n)^2}{2}\frac{f^{''}(c_n)}{f^{'}(x_n)}$$

$$x_n - \frac{f(x_n)}{f^{'}(x_n)} - \alpha = \frac{(\alpha - x_n)^2}{2}\frac{f^{''}(c_n)}{f^{'}(x_n)}$$

$$x_{n+1} - \alpha = \frac{1}{2}(\alpha - x_n)^2 \frac{f^{''}(c_n)}{f^{'}(x_n)}$$

We know that by definition, $x_n - \alpha = e_n$, and that $\lim_{n\to\infty} x_n = \alpha$, so this can be rewritten as

$$\frac{e_{n+1}}{{e_n}^2} = \frac{1}{2}\frac{f^{''}(\alpha)}{f^{'}(\alpha)}$$

Because the right hand side is constant, and the ratio of errors is of the form $\frac{R}{R^2} = c$, we can say that the sequence is quadratically convergant.

If a poor starting guess is chosen, Newton's method will not converge at all towards a root of the function. A “bad” starting guess would be in the region [a,b] about the root such that

- $f, f^{'}, f^{''}$ does not exist in [a,b]
- $f^{'} = 0$ in [a,b]
- $f^{''}$ changes sign in [a,b]

Therefore, **ii and iii** are correct.

(d) True. If an initial guess for Newton's root finding method is to be considered "good", it must follow some criteron.

- $f, f^{'}, f^{''}$ must exist exist
- Choose a, b such that $f(a) * f(b) < 0$
- $f^{'} \neq 0$ in $[a, b]$
- $f^{''}$ does not change sign in $[a, b]$
- $|\frac{f(a)}{f^{'}(a)}|$ and $|\frac{f(b)}{f^{'}(b)}| < |b - a|$

If all of these hold true, the there will be a unique root in $[a, b]$, and any starting guess in $[a, b]$ will converge to the root. Therefore, if x_0 leads to convergance, then any guess between x^* and x_0 will also lead to convergance.

(e) One of the main advantages of Newton's method over the bisection method is the speed at which it can approximate roots. The quadratic convergance rate of Newton's method means that given a good starting guess, it can approximate a root in a fraction of the time of the bisection method.

However, unlike the bisection method, it requires a good starting guess. If the initial zero chosen does not follow some precise specifications (see part(d)), then the method will not converge. Despite its much longer runtime, the bisection method will always work, given that α lies in the initial range.

Also, Newton's method can be somewhat more difficult to compute, as the $f^{'}$ is needed to compute the root of f. The bisection method relies solely on f.

(f) $f(x) = 2 - x^2$
$f^{'}(x) = -2x$
We choose $x_0 = 2$ for a starting guess.

The rule for Newton's method is $x_{n+1} = x_n - \frac{f(x_n)}{f^{'}(x_n)}$

$x_1 = x_0 - \frac{f(x_0)}{f^{'}(x_0)} = \frac{3}{2}$

$x_2 = x_1 - \frac{f(x_1)}{f^{'}(x_1)} = \frac{17}{12}$

$x_3 = x_2 - \frac{f(x_2)}{f^{'}(x_2)} = \frac{577}{408}$

$x_4 = x_3 - \frac{f(x_3)}{f^{'}(x_3)} = \frac{665857}{470832}$

$x_5 = x_4 - \frac{f(x_4)}{f^{'}(x_4)} = \frac{886731088897}{627013566048}$

At this point, if we define the tolerance as 10^{-12}, we must see if we are close enough to the root. Because technically we do not know the true answer $\sqrt{2}$, we see if the last two roots are sufficiently close together.
$x_5 - x_4 = \frac{665857}{470832} - \frac{886731088897}{627013566048} = \frac{1}{627013566048} \approx 1.6 * 10^{-12}$

The result is within the defined tolerance, and the algorithm is complete, with the calculated root being

$$x_5 = \frac{886731088897}{627013566048} \approx 1.41421356237468$$

Compared to the actual value of the root, $\sqrt{2}$, the relative error is $\frac{x_5 - \sqrt{2}}{\sqrt{2}} \approx 1.13 * 10^{-12}$